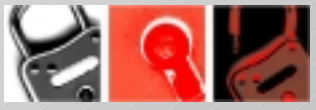
Developing a Customized STEP Implementation

in the context of PDM application domain



Dr.-Ing. Anna Wasmer
PDTec GmbH
D-76131 Karlsruhe, Germany
Fon: +49-721-9658-675
Fax: +49-721-9658-263
E-Mail: wasmer@pdtec.de
Web: www.pdtec.de

PDTec. Enabling Product Data Technology

NASA STEP Workshop, Pasadena, CA, January 16/19, 2001

PDTec.

Challenges of eEngineering Communication

- **eEngineering requires high degree of communication**
  - Concurrent/ simultaneous engineering
  - OEM - Supplier collaboration
  - Use of catalogue/standard parts
- **Need for expanded life cycle management through**
  - Strategic partnerships
  - Development cooperations
  - Inclusion of key suppliers into business workflow
  - Global environment
- **Exchange of 'just' geometry is not sufficient**
  - Exchangeability of product definitional information (e.g. product identification) and organizational information (e.g. approval) required
- **Data managed by PDM systems is a basis for eEngineering communication**

PDM Customization

- **Commercial PDM systems often are toolkits that need to be customized**
  - Customer specific data model
    - » Adapt/extend the PDM standard data model
    - » Build / implement own data model
  - Customer specific scope and constraints
    - » Terminology
    - » Attribute value domains (e.g. names of approval or lifecycle states)
    - » Part numbering systems
  - Customer specific business practices and processes , e.g.
    - » Implement processes to support their specific business practices (object lifecycles, roles, signatures, ..)
    - » Use of assembly structure or document structure to describe geometric relationships and transformation matrices between parts
  - Customized user interface

Need for Customizable Processors - 1

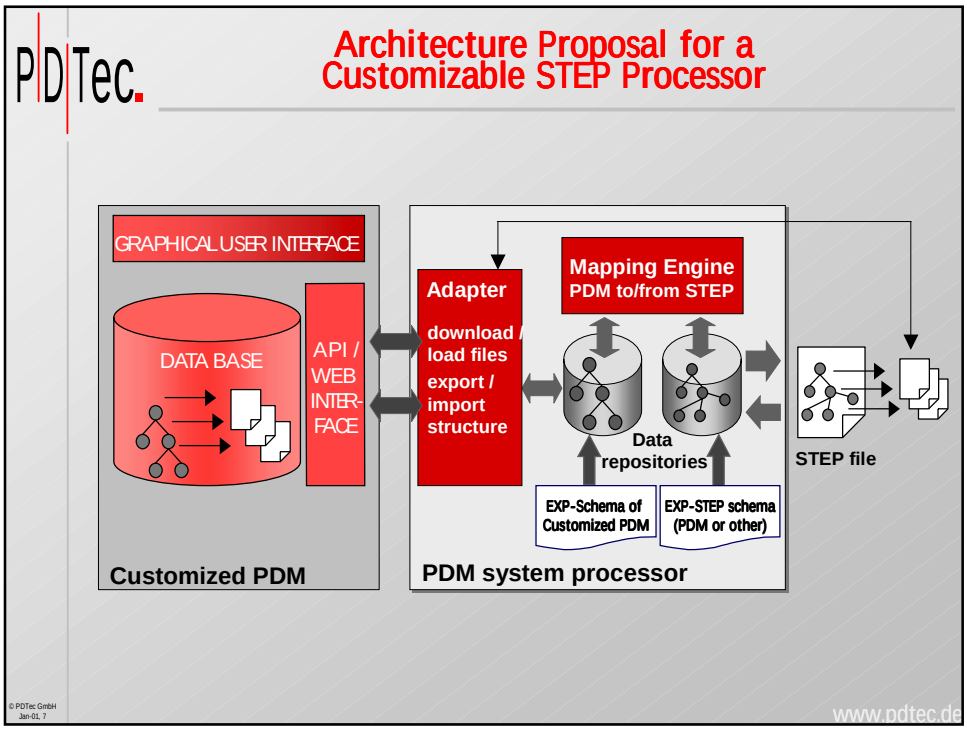
- **Initial processor customization according to system customization**
- **Continuous evolution and change of implementations**
  - E.g. new data types, attributes and attribute value sets
- **Different mapping „variations“ may be necessary to communicate with other applications (e.g. PDM, ERP, CAD,..) and organizations**
  - Different instantiation practices
  - Bilaterally agreed attribute value mappings
  - Support different views of the product, e.g. as designed, as built

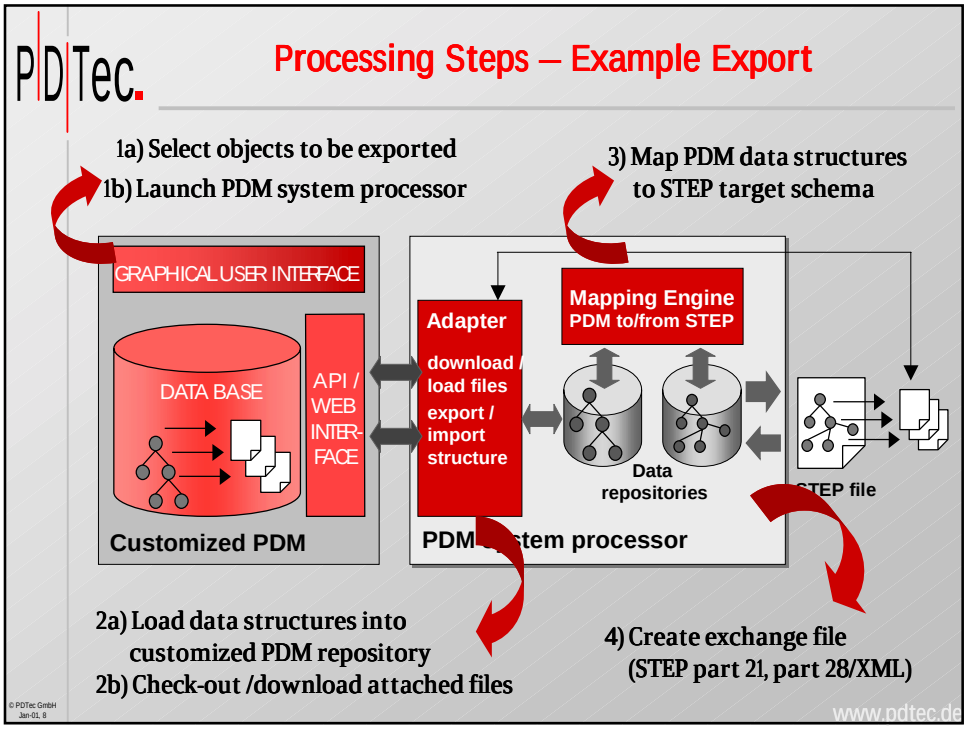
Need for Customizable Processors - 2

- **Evolution and change of supported standards**
  - e.g. new revisions, migration to modules approach
- **Ability to use and support selection of different standards and target data models**
  - Different set of target modules or schemas (e.g. STEP APs)
  - Different representation format of target data set (e.g. STEP part 21 or part 28/XML)
- **Specific system environments**
  - e.g. network distribution, client-server, web based access, integration with exchange tools

Consequences

- **System vendors / processor implementors**
  - Standard system processors should be easily customizable
    - » Customer specific data model, scope and constraints
    - » Customer specific business practices and processes
    - » Evolution and change of supported standards
    - » Migration path to new standards (data representation formats and target data models)
- **User companies**
  - Use implementation technology that provide
    - » Good support for initial processor development/customization
    - » Good support for continuous evolution of processor according to PDM implementation phases
    - » Flexibility to support different exchange scenarios and business relationships
    - » Possibility to include different mapping „variations“ to optimize exchange with selected partners
    - » Easy integration in changing system environments


PDTec.
Architecture Proposal for a Customizable STEP Processor
GRAPHICAL USER INTERFACE
DATA BASE
API / WEB INTER-FACE
Customized PDM
Adapter
download / load files
export / import structure
Mapping Engine PDM to/from STEP
Data repositories
EXP-Schema of Customized PDM
EXP-STEP schema (PDM or other)
PDM system processor
STEP file
© PDTec GmbH Jan-01, 7
www.pdtec.de


PDTec.
Processing Steps – Example Export
1a) Select objects to be exported
1b) Launch PDM system processor
3) Map PDM data structures to STEP target schema
GRAPHICAL USER INTERFACE
DATA BASE
API / WEB INTER-FACE
Customized PDM
Adapter
download / load files
export / import structure
Mapping Engine PDM to/from STEP
Data repositories
STEP file
2a) Load data structures into customized PDM repository
2b) Check-out /download attached files
4) Create exchange file (STEP part 21, part 28/XML)
© PDTec GmbH Jan-01, 8
www.pdtec.de

Steps for Developing a Customized STEP Processor

(1) **Create data repositories**
- Identify STEP target data models (PDM schema, APxxx, set of modules)
- Generate EXPRESS representation of customized PDM data model
  - » e.g. tool for automatic extraction

(2) **Develop system adapter**
- Define business practices and rules depending on specific user settings
  - » e.g. when to perform certain actions (e.g. check-in/-out, replace, add, etc.)
- Business Logic implementation
- Import /export data of data structures and attached files via API

(3) **Develop Mapping engine**
- Mapping specification (EXPRESS-X)
- Definition of extensible configuration tables e.g. attribute value mappings, bilateral agreements, etc.
- Automatic generation of mapping engine

(4) **Additional functionality (e.g. PDM GUI extensions for data selection, browsing and editing. checking modules)**

➨ **Steps (2), (3), and (4) can be parallelized**

The Mapping Language EXPRESS-X

- **Structural data mapping language to**
  - Allow an unambiguous specification of the relationship between models defined in EXPRESS (ISO 10303-11)
  - Support mapping of data defined by one EXPRESS model to data defined by another EXPRESS model
- **Defined as extension of the EXPRESS language**
- **Available as part 14 of the SC4/STEP series standards**
- **Committee draft (CD) successfully balloted (Dec. 2000)**
- **10303-14 will go forward for registration as a DIS (Draft International Standard), after ballot comment resolution**
- **Generation of EXPRESS-X statements from textual mapping specifications (e.g. mapping tables)**

EXPRESS-X Fundamental Principles

- **Explicit specification of one-way mappings between n source and m target EXPRESS schemas**
  - Inverse mappings contained in different mapping specification
- **Modular structure of mappings**
  - Mappings can be REFERENCEd like EXPRESS SCHEMAs
  - Mappings can be configured and extended depending on exchange needs with selected partners
- **Declarative approach of the specification**
  - Readability
  - Same level of abstraction as mapped data models
  - Independence of execution order
- **Procedural extensions**
  - Inclusion of predefined algorithms on source and target data
- ➨ **Clear separation between description and execution model**

Structure of an EXPRESS-X Specification

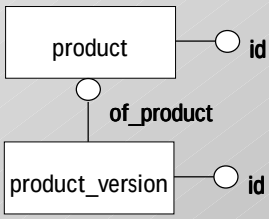
- **Identification of source and target schemas**
- **Use of externally defined mappings and functions**
- **(Data-) Type mapping**
- **Declaration of constants (EXPRESS syntax) and target/view instances which are not directly related to source data**
- **Declaration of procedures and functions (EXPRESS syntax)**
- **MAP and VIEW declarations**
  - MAP
    specification of the mapping between pre-existing source and target schemas (entities)
  - VIEW
    definition of a view schema (view entities) derived from one or more source schemas (source entities)

PDTec.

EXPRESS-X Example

SOURCE SCHEMA:

TARGET SCHEMA:



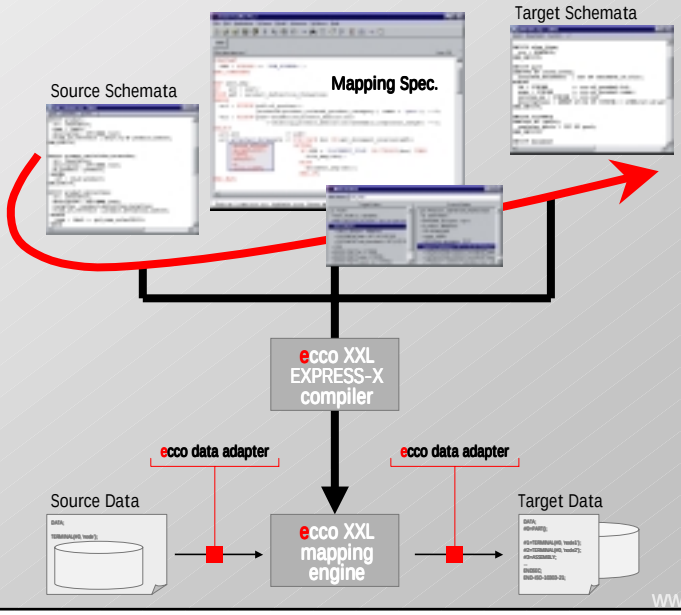
EXPRESS-X SCHEMA:

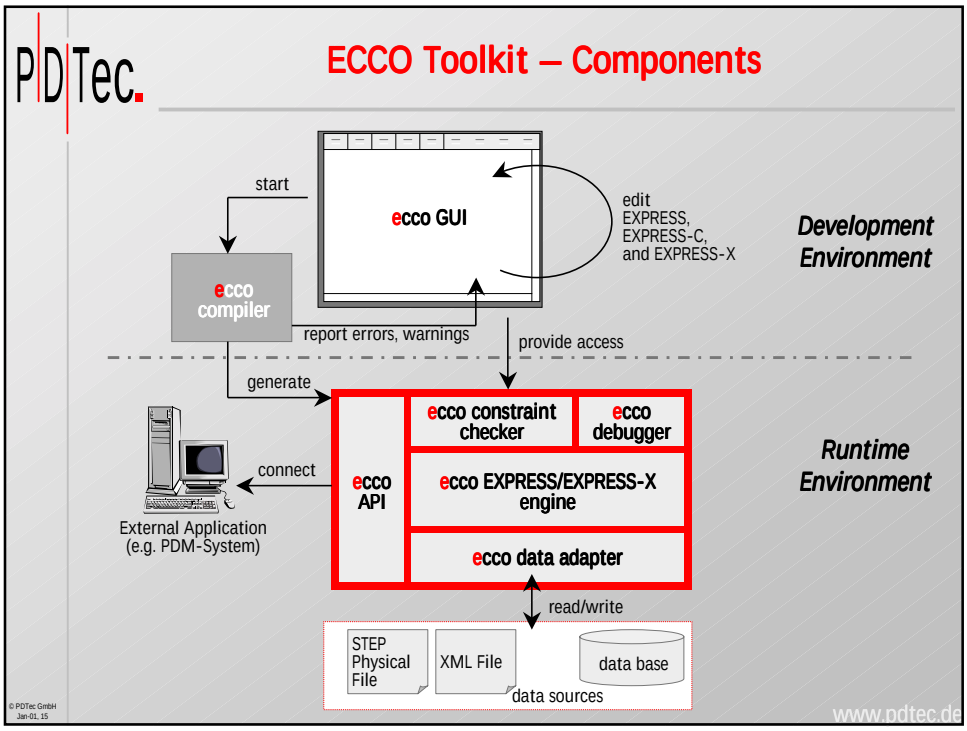
```
MAP product_map AS
  p : product;
FROM
  prt : part;
SELECT
  p.id := prt.partno;
END_MAP;
```

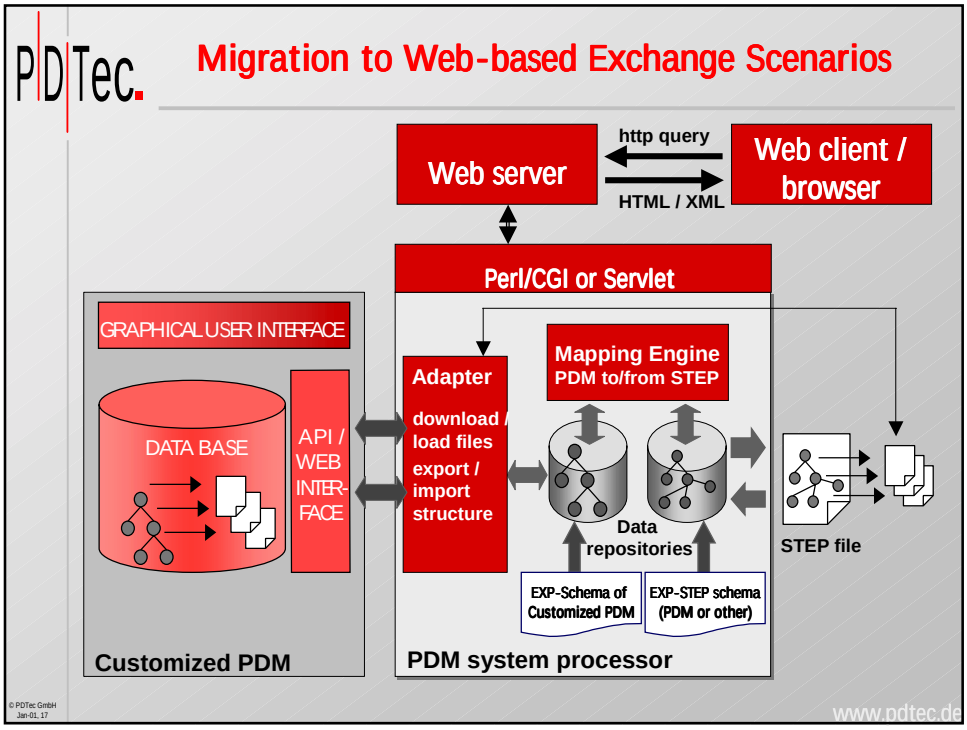
```
MAP product_version_map AS
  pv : product_version;
FROM
  prt : part;
SELECT
  pv.id := prt.version;
  of_product := product_map(prt);
END_MAP;
```


PDTec.

Mapping Engine Generation using ECCO XXL




PDTec.
ECCO Toolkit – Components
start
ecco GUI
edit
EXPRESS,
EXPRESS-C,
and EXPRESS-X
Development
Environment
ecco
compiler
report errors, warnings
provide access
generate
ecco constraint
checker
ecco
debugger
ecco
API
ecco EXPRESS/EXPRESS-X
engine
ecco data adapter
connect
External Application
(e.g. PDM-System)
Runtime
Environment
read/write
STEP
Physical
File
XML File
data base
data sources
© PDTec GmbH
Jan-01, 15
www.pdtec.de


PDTec.
Migration to Web-based Exchange Scenarios
Web server
http query
HTML / XML
Web client /
browser
Perl/CGI or Servlet
GRAPHICAL USER INTERFACE
Adapter
download /
load files
export /
import
structure
Mapping Engine
PDM to/from STEP
DATA BASE
API /
WEB
INTER-
FACE
Data
repositories
STEP file
EXP-Schema of
Customized PDM
EXP-STEP schema
(PDM or other)
Customized PDM
PDM system processor
© PDTec GmbH
Jan-01, 17
www.pdtec.de

PDTec.

Conclusions

- **Advantages of system adapter and mapping engine as separate processor components**
  - Changes in system API's or technology replacement (e.g. web based interface) do not affect mapping engine
  - Reuse of sys. adapter for mappings to different target schemas
    - Support of additional standards (STEP data models, or others)
  - Use of EXPRESS-X reduces development time for data mappings and allows easy adaptation of mapping engine
  - Independence of data representation format on target side
    - Changes in target schema/ mapping do not affect system adapter
  - Parallelization of development and customization work
    - use of different experts
  - Distribution in the network
  - Easy migration to new technology components (e.g. web)
  - Easy integration of additional development tools (schema extractor, schema comparison, graphical mapping tool)